EXACT AND APPROXIMATE SOLUTIONS OF FUZZY LR LINEAR SYSTEMS: NEW ALGORITHMS USING A LEAST SQUARES MODEL AND THE ABS APPROACH

R. GHANBARI, N. MAHDAVI-AMIRI AND R. YOUSEFPOUR

Abstract. We present a methodology for characterization and an approach for computing the solutions of fuzzy linear systems with LR fuzzy variables. As solutions, notions of exact and approximate solutions are considered. We transform the fuzzy linear system into a corresponding linear crisp system and a constrained least squares problem. If the corresponding crisp system is incompatible, then the fuzzy LR system lacks exact solutions. We show that the fuzzy LR system has an exact solution if and only if the corresponding crisp system is compatible (has a solution) and the solution of the corresponding least squares problem is equal to zero. In this case, the exact solution is determined by the solutions of the two corresponding problems. On the other hand, if the corresponding crisp system is compatible and the optimal value of the corresponding constrained least squares problem is nonzero, then we characterize approximate solutions of the fuzzy system by solution of the least squares problem. Also, we characterize solutions by defining an appropriate membership function so that an exact solution is a fuzzy LR vector having the membership function value equal to one and, when an exact solution does not exist, an approximate solution is a fuzzy LR vector with a maximal membership function value. We propose a class of algorithms based on ABS algorithm for solving the LR fuzzy systems. The proposed algorithms can also be used to solve the extended dual fuzzy linear systems. Finally, we show that, when the system has more than one solution, the proposed algorithms are flexible enough to compute special solutions of interest. Several examples are worked out to demonstrate the various possible scenarios for the solutions of fuzzy LR linear systems.

1. Introduction

Linear systems arise from many areas of science and engineering. For real world problems, some parameters of a linear system may be uncertain. Therefore, various types of fuzzy linear systems and their solutions have attracted much interest in recent years. An approach for solving a particular fuzzy linear system (FLS) would often be specific to the way the solution is characterized. Solutions of several kinds of fuzzy linear systems under various assumptions have been proposed in recent years (see [3], [4], [6], [10], [11], [13], [18], [23], [24], as examples).

Received: October 2008; Revised: June 2009; Accepted: October 2009

Key words and phrases: Fuzzy linear system, Fuzzy LR solution, ABS algorithm, Least squares approximation.

A particular type of fuzzy linear systems is defined to be:

$$A\tilde{x} = \tilde{b}, \tag{1}$$

where the coefficient matrix $A \in \mathbb{R}^{m\times n}$, $m \leq n$, is crisp, $\tilde{b}_{m\times 1}$ and $\tilde{x}_{n\times 1}$ are fuzzy vectors. Friedman et al. [10] use an embedding method for solving an special case of (1) when A is a square ($m = n$) nonsingular matrix. They introduce new notions for solutions of a square FLS (strong and weak solutions) and transform the FLS to a crisp square system. In their method, the inverse matrix is used to solve the FLS.

Here, we develop an embedding method of fuzzy LR linear system (FLRLS; fuzzy linear system with LR fuzzy variable and LR fuzzy right hand side) for the system, in its general setting, where $A \in \mathbb{R}^{m\times n}$, $m \leq n$, $\tilde{b}_{m\times 1}$ and $\tilde{x}_{n\times 1}$ are LR fuzzy vectors (see Definition 2.13). Theses types of systems are of particular interest, as the LR fuzzy numbers may be considered for use in real world problems (see [4, 7, 15, 16, 19, 25]).

First, we develop an adaptation of the approach in Friedman et al. [10] to nonsquare LR systems. Then, we introduce a new notion of (exact or approximate) solution for the FLRLS after transforming the FLRLS into a corresponding crisp linear system and a constrained least squares problem. We show that if the corresponding crisp system is incompatible, then FLRLS lacks any exact solution; in such a case, we do not consider any approximate solution for the system either. We further show that the fuzzy LR system has an exact solution if and only if the corresponding crisp system is compatible and the solution of the corresponding least squares problem is equal to zero. In this case, an exact solution of the FLRLS can be computed using the extension principal (if $m = n$ and A is a square nonsingular matrix, then the exact solution computed by our method is equivalent to the strong solution computed by Friedman et al. [10]). On the other hand, if the corresponding crisp system is compatible and the optimal value of the corresponding constrained least squares problem is nonzero, then no exact solution exists. In this situation, it is reasonable to seek approximate solutions, which, in some meaningful sense, would minimize the residue of the systems of equations. For crisp systems, the need for such approximate solutions arise from many "data fitting" applications and our suggested approach for computing such solutions is widely practiced. The most common model for minimization of the residue is the least squares model (see [5] and [14] for excellent treatments of the least squares model).

Thus, here, we introduce and compute approximate solutions of FLRLS when an exact solution dose not exist. It should be pointed out that an idea of using a constrained least squares problem was also used by Abramovich et al. [4] on a different type of FLS.

There are many methods for solving the corresponding crisp linear system. For several reasons, we use an ABS algorithm to find all solutions of the corresponding crisp linear system. Since there is a variety of the ABS algorithms within ABS class, then our approach would offer a variety of algorithmic techniques for various possible versions of the FLRLS. Moreover, when there are more than one solution, then we will see that certain specific solutions of interest may be computed using

a particular algorithm within the class.
ABS methods were first introduced by Abaffy et al. [1], based upon Egervary's rank reducing algebraic process [8]. ABS methods comprise a large class of methods that are used for solving linear algebraic systems, nonlinear algebraic equations, linear least squares problems, optimization problems and Diophantine equations (see [2], [9], [12], [20, 21, 22]). Using a particular ABS method in our proposed algorithm, we would be able to adapt the algorithm to solve special cases of FLS to obtain a specific solution as well (e.g., solution with the least Euclidean norm in the solution space; see Section 2.2). Also, with a small modification, we would be able to extend our proposed algorithm to solve an extended dual fuzzy linear system [13].
In Section 2, we give the necessary definitions, some properties of fuzzy arithmetic and basic aspects of ABS algorithms. We introduce the least squares model and develop our proposed algorithm with the corresponding results and present relevant discussions in Section 3. We conclude in Section 4.

2. Preliminaries

Here, we give some basic definitions and results concerning fuzzy numbers in Section 2.1, followed by a brief description of basic steps of the ABS algorithms for solving liner equations in Section 2.2.

2.1. **Fuzzy Numbers.** A fuzzy number is a fuzzy quantity A (see [19]) that represents a generalization of a real number r. Intuitively, $\mu_A(x)$ should be a measure of how good $\mu_A(x)$ "approximates" r, and certainly one reasonable requirement is that $\mu_A(r) = 1$ [19].

Definition 2.1. [19] A fuzzy number is a fuzzy quantity A that satisfies the following conditions:

(1) $\mu_A(x) = 1$ for exactly one x.
(2) The support $\{x : \mu_A(x) > 0\}$ of A is bounded.
(3) The α-cuts of A are closed intervals.

Remark 2.2. There are several other definitions for fuzzy number in the literature (see [25]) which differ from the above definition.

Proposition 2.3. The followings hold [19]:

(1) Real numbers are fuzzy numbers.
(2) A fuzzy number is a convex fuzzy quantity.
(3) A fuzzy number is upper semicontinuous.
(4) If A is a fuzzy number with $\mu_A(r) = 1$, then μ_A is nondecreasing on $(-\infty, r\,]$ and nonincreasing on $(r, \infty\,]$.

Theorem 2.4. [19] *If A and B are fuzzy numbers, then so are $A + B$ and $-A$, with $A + B$ and $-A$ as defined in* [19].

Definition 2.5. [7] The decreasing map $L : \mathbb{R}^+ \to [0, 1]$ is a shape function if the followings hold:

$$\begin{cases} L(0) = 1, \\ L(1) = 0, \\ 0 \le L(x) \le 1, \quad x \ne 0, 1. \end{cases}$$

Definition 2.6. A fuzzy number $\tilde{A}$ is of LR-type if there exist shape functions L (for left), R (for right) and scalars $\alpha > 0$, $\beta > 0$ with

$$\mu_{\tilde{A}}(x) = \begin{cases} L\left(\frac{a-x}{\alpha}\right), & x \leq a, \\ R\left(\frac{x-a}{\beta}\right), & x \geq a. \end{cases}$$

The mean value of $\tilde{A}$, a, is a real number, and α, β are called the left and right spreads respectively. $\tilde{A}$ is denoted by $(a, \alpha, \beta)_{LR}$. A crisp number "a" is specified to be $(a, 0, 0)_{LR}$.

Definition 2.7. A fuzzy number $\tilde{A} = (m, \alpha, \alpha)_{LL}$ is a symmetric fuzzy number.

Remark 2.8. According to Definition 2.1, throughout the paper, we assume that all the supports of LR-type fuzzy numbers are bounded.

Note that there are many candidates for a shape function; for example, $\max(0, 1-x)^p$ and $\max(0, 1-x^p)$, where $p > 0$.

Example 2.9. Let $\tilde{M} = (0, 1, 2)_{LR}$, $L = \max(0, 1-x)$ and $R = \max(0, 1-x^2)$. Then,

$$\mu_{\tilde{M}}(x) = \begin{cases} x+1, & x \in [-1, 0], \\ 1 - \frac{x^2}{4}, & x \in [0, 2]. \end{cases}$$

Definition 2.10. $\tilde{M} = (m, \alpha, \beta)_{LR}$ is a triangular fuzzy number if $L = R = \max(0, 1-x)$.

Theorem 2.11. [25] *Let $\tilde{M} = (m, \alpha, \beta)_{LR}$, $\tilde{N} = (n, \gamma, \delta)_{LR}$ and $\lambda \in \mathbb{R}^+$. Then,*

(1) $\lambda\tilde{M} = (\lambda m, \lambda\alpha, \lambda\beta)_{LR}$.
(2) $-\tilde{M} = (-m, \beta, \alpha)_{LR}$.
(3) $\tilde{M} \bigoplus \tilde{N} = (m+n, \alpha+\gamma, \beta+\delta)_{LR}$.

Note 2.12. Here, we denote the set of LR fuzzy numbers by $\mathbb{F}\left(\mathbb{R}^1\right)_{LR}$.

Definition 2.13. $\tilde{x}=(\tilde{x}_1, \tilde{x}_2, \ldots, \tilde{x}_n)^T$, denoted by $\tilde{x} \in \mathbb{F}(\mathbb{R}^n)_{LR}$, is called a fuzzy vector where, $\tilde{x}_i \in \mathbb{F}(\mathbb{R}^1)_{LR}, i = 1, \ldots, n$.

2.2. **Basic ABS Algorithms.** ABS methods, in their general forms, have been developed by Abaffy, Broyden and Spedicato [1]. Consider the system of linear equations

$$Ax = b, \tag{2}$$

where $A \in \mathbb{R}^{m \times n}, b \in \mathbb{R}^m$ and $rank(A) = m$. Let $A = (a_1, \ldots, a_m)^T, a_i \in \mathbb{R}^n, i = 1, \ldots, m$ and $b = (b_1, \ldots, b_m)^T$. Also, let $A_i = (a_1, \ldots, a_i)$ and $b^{(i)} = (b_1, \ldots, b_i)^T$. Assume $x_1 \in \mathbb{R}^n$ arbitrary and $H_1 \in \mathbb{R}^{n \times n}$, Spedicato's parameter, arbitrary and nonsingular. Note that for any $x \in \mathbb{R}^n$ we can write $x = x_1 + H_1^T q$ for some $q \in \mathbb{R}^n$. The ABS class of methods contains direct iterative methods for computing the general solution of (2). In the beginning of the ith iteration, $i \geq 1$, the general solution of the first $i-1$ equation is at hand. We realize that if x_i is a solution for

the first $i-1$ equations and if $H_i \in \mathbb{R}^{n\times n}$, with $rank(H_i) = n-i+1$, is so that the columns of H_i^T span the null space of A_{i-1}^T, then

$$x = x_i + H_i^T q$$

with arbitrary $q \in \mathbb{R}^n$, forms the general solution of the first $i-1$ equations. That is, with

$$H_i A_i = 0,$$

we have,

$$A_{i-1}^T x = b^{(i-1)}.$$

Now, since $rank(H_i) = n-i+1$ and H_i^T is a spanning matrix for $null(A_i^T)$ by assumption (one that is trivially valid for $i=1$), then letting

$$p_i = H_i^T z_i,$$

with arbitrary $z_i \in \mathbb{R}^n$, Broyden's parameter, we have $A_{i-1}^T p_i = 0$ and

$$x(\alpha) = x_i - \alpha p_i,$$

for any scalar α solves the first $i-1$ equations. We can set $\alpha = \alpha_i$ so that $x_{i+1} = x(\alpha_i)$ solves the ith equation as well. If we let

$$\alpha_i = \frac{a_i^T x_i - b_i}{a_i^T p_i},$$

with assumption $a_i^T p_i \neq 0$, then

$$x_{i+1} = x_i - \alpha_i p_i,$$

is a solution for the first i equations. Now, to complete the ABS step, H_i must be updated to H_{i+1} so that $H_{i+1}A_i = 0$. It will suffice to let

$$H_{i+1} = H_i - u_i v_i^T, \tag{3}$$

and select u_i and v_i so that $H_{i+1}a_j = 0, j = 1,\ldots,i$. The updating formula (3) for H_i is a rank-one correction to H_i. The matrix H_i is generally known as the Abaffian. The ABS methods usually use $u_i = H_i a_i$ and $v_i = H_i^T w_i / w_i^T H_i a_i$, where w_i, Abaffy's parameter, is an arbitrary vector satisfying

$$w_i^T H_i a_i \neq 0.$$

Thus, the updating formula can be written as:

$$H_{i+1} = H_i - \frac{H_i a_i w_i^T H_i}{w_i^T H_i a_i}.$$

We now give the general steps of an ABS algorithm [1, 2]. We have given the steps of the algorithms, since, later on, we make specific use of the parameters for computing special solutions from amongst the general solution. Below, r_{i+1} denotes the rank of A_i and hence the rank of H_{i+1} equals $n - r_{i+1}$.

We note that after the completion of the algorithm, the general solution of (2), if compatible, is written as $x = x_{m+1} + H_{m+1}^T q$, where $q \in \mathbb{R}^n$ is arbitrary. Below, we list certain properties of the ABS methods [2].

Algorithm 1 ABS Algorithm

Step 1: Choose $x_1 \in \mathbb{R}^n$, arbitrary, and $H_1 \in \mathbb{R}^{n\times n}$, arbitrary and nonsingular. Set $i = 1$, $r_1 = 0$.

Step 2: Compute $\tau_i = a_i^T x_i - b_i$ and $s_i = H_i a_i$.

Step 3: **If** ($s_i = 0$ **and** $\tau_i = 0$) **then** let $x_{i+1} = x_i$, $H_{i+1} = H_i$, $r_{i+1} = r_i$ and **go to Step** 7 (the ith equation is redundant). **If** ($s_i = 0$ **and** $\tau_i \neq 0$) **then Stop** (the ith equation and hence the system is incompatible).

Step 4: $\{s_i \neq 0\}$ Compute the search direction $p_i = H_i^T z_i$, where $z_i \in \mathbb{R}^n$ is an arbitrary vector satisfying $z_i^T H_i a_i = z_i^T s_i \neq 0$. Compute $\alpha_i = \tau_i / a_i^T p_i$ and let $x_{i+1} = x_i - \alpha_i p_i$.

Step 5: Update H_i to H_{i+1} by:

$$H_{i+1} = H_i - \frac{H_i a_i w_i^T H_i}{w_i^T H_i a_i},$$

where $w_i \in \mathbb{R}^n$ is an arbitrary vector satisfying $w_i^T s_i \neq 0$.

Step 6: Let $r_{i+1} = r_i + 1$.

Step 7: **If** $i = m$ **then Stop** (x_{m+1} is a solution) **else** let $i = i + 1$ and **go to Step** 2.

- The direction search vectors $p_1, \ldots, p_i$ are linearly independent.
- The set of directions $p_1, \ldots, p_i$ together with independent columns of H_{i+1}^T form a basis for $\mathbb{R}^n$.
- $H_i a_i \neq 0$ if and only if a_i is linearly independent of $a_1, \ldots, a_{i-1}$. Equivalently, $H_i a_i = 0$ if and only if a_i is linearly dependent of $a_1, \ldots, a_{i-1}$.
- $Rank(H_{i+1}) = n - r_{i+1}$, where $r_{i+1} = rank(A_i)$.
- Since $rank(A) = r_{m+1}$ then the general solution of (2) is written as $x = x_{m+1} + H^T q$, where $H^T_{n\times(n-r_{m+1})}$ is a matrix with $n - r_{m+1}$ independent columns and $q \in \mathbb{R}^{n-r_{m+1}}$ is arbitrary. Note that since $w_i^T H_{i+1} = 0$, then any row of H_{i+1} corresponding to a nonzero component of w_i is a dependent row and can be removed in **Step** 5. The matrix H is the final Abaffian obtained by removal of a dependent row at every step.
- The Huang's algorithm [2] is a special ABS algorithm with, Spedicato's parameter, $H_1 = I$ in **Step** 1, $z_i = a_i$ in **Step** 4, $w_i = a_i / a_i^T H_i a_i$. If, in addition, we let $x_1 = \lambda a_1$ where λ is a scalar then the sequence of iterates $x_2, \ldots, x_{m+1}$ generated by Algorithm 1 has the minimal Euclidean norm amongst all solutions of the corresponding subsystems [2]. Also, we have the same property when we use Algorithm 1 by setting $H_1 = I$ in **Step** 1, $z_i = H_i a_i$ in **Step** 4, $w_i = z_i / z_i^T z_i$ (resulting in the modified Huang's algorithm; see [2]) and $x_1 = \lambda a_1$ where λ is a scalar. It is shown that the modified Huang's algorithm is more stable numerically than Huang's algorithm [2].

3. Fuzzy LR Linear Systems

In many applications, fuzzy LR numbers are used to describe the models, because fuzzy LR numbers increase computational efficiency without limiting the generality beyond acceptable limits [25]. Thus, we focus on solving fuzzy LR systems as defined below.

Definition 3.1. The system,

$$A\tilde{x} = \tilde{b}, \tag{4}$$

where $A = [a_{ij}]_{m\times n} \in \mathbb{R}^{m\times n}$, $m \leq n$, and $\tilde{b} \in \mathbb{F}(\mathbb{R}^m)_{LR}$, an LR fuzzy vector, are given and $\tilde{x} \in \mathbb{F}(\mathbb{R}^n)_{LR}$ is an unknown LR fuzzy vector (of the same LR type as $\tilde{b}$) to be found, is called a fuzzy LR linear system (FLRLS).

In Definition 3.1, let $\tilde{b} = \left(\tilde{b}_1, \ldots, \tilde{b}_m\right)^T$, $\tilde{x} = (\tilde{x}_1, \ldots, \tilde{x}_n)^T$ and a_i^T be the ith row of A. We can reformulate (4) in the form,

$$a_i^T \tilde{x} = \tilde{b}_i,\ i = 1, \ldots, m,$$

or,

$$\sum_{j=1}^{n} a_{ij}\tilde{x}_j = \tilde{b}_i, \quad i = 1, \ldots, m. \tag{5}$$

Let $B_i^- = \{j : a_{ij} < 0\}$ and $B_i^+ = \{j : a_{ij} \geq 0\}$. Then, from (5) we have,

$$\sum_{j\in B_i^-} a_{ij}\tilde{x}_j + \sum_{j\in B_i^+} a_{ij}\tilde{x}_j = \tilde{b}_i, \quad i = 1, \ldots, m. \tag{6}$$

Let $\tilde{x}_j = (x_j, \alpha_j, \beta_j)_{LR}$. Then, we can write (6) as follows:

$$\sum_{j\in B_i^+} a_{ij}\,(x_j, \alpha_j, \beta_j)_{LR} + \sum_{j\in B_i^-} a_{ij}\,(x_j, \alpha_j, \beta_j)_{LR} = \tilde{b}_i, \quad i = 1, \ldots, m.$$

From Theorem 2.11, we obtain,

$$\left(\sum_{j\in B_i^+} a_{ij}x_j, \sum_{j\in B_i^+} a_{ij}\alpha_j, \sum_{j\in B_i^+} a_{ij}\beta_j\right)_{LR} + \left(\sum_{j\in B_i^-} a_{ij}x_j, -\sum_{j\in B_i^-} a_{ij}\beta_j, -\sum_{j\in B_i^-} a_{ij}\alpha_j\right)_{LR}$$
$$= \tilde{b}_i, \quad i = 1, \ldots, m.$$

Let $\tilde{b}_i = \left(b_i, b_i^l, b_i^r\right)_{LR}$. Then,

$$\left(\sum_{j\in B_i^+} a_{ij}x_j + \sum_{j\in B_i^-} a_{ij}x_j, \sum_{j\in B_i^+} a_{ij}\alpha_j - \sum_{j\in B_i^-} a_{ij}\beta_j, \sum_{j\in B_i^+} a_{ij}\beta_j - \sum_{j\in B_i^-} a_{ij}\alpha_j\right)_{LR} = \left(b_i, b_i^l, b_i^r\right)_{LR}, \quad i = 1, \ldots, m. \tag{7}$$

So, we must have,

$$\begin{cases} \sum_{j=1}^{n} a_{ij}x_j = b_i, & i = 1, \ldots, m, \\ x_j \in \mathbb{R}, & j = 1, \ldots, n, \end{cases} \tag{8}$$

and simultaneously,

$$\begin{cases} \sum_{j\in B_i^+} a_{ij}\alpha_j - \sum_{j\in B_i^-} a_{ij}\beta_j = b_i^l, & i = 1, \ldots, m, \\ \sum_{j\in B_i^+} a_{ij}\beta_j - \sum_{j\in B_i^-} a_{ij}\alpha_j = b_i^r, & i = 1, \ldots, m, \\ \alpha_j, \beta_j > 0, & j = 1, \ldots, n. \end{cases} \tag{9}$$

Therefore, the system (4) is equivalent to the crisp linear systems (8) and (9). Now, define two matrices,

$$[B^+]_{ij} = \begin{cases} a_{ij}, & a_{ij} \geq 0, \\ 0, & a_{ij} < 0, \end{cases} \qquad [B^-]_{ij} = \begin{cases} a_{ij}, & a_{ij} < 0, \\ 0, & a_{ij} \geq 0, \end{cases}$$

for all $i = 1, \ldots, m$ and $j = 1, \ldots, n$. Also, let $x = (x_1, \ldots, x_n)^T, b = (b_1, \ldots, b_m)^T$, $\alpha = (\alpha_1, \ldots, \alpha_n)^T$, $\beta = (\beta_1, \ldots, \beta_n)^T$, $b^l = (b_1^l, \ldots, b_n^l)^T$ and $b^r = (b_1^r, \ldots, b_n^r)^T$. Then, the systems (8) and (9) are respectively written in the matrix forms,

$$Ax = b, \tag{10}$$

and

$$\begin{bmatrix} B^+ & -B^- \\ -B^- & B^+ \end{bmatrix} \begin{bmatrix} \alpha \\ \beta \end{bmatrix} = \begin{bmatrix} b^l \\ b^r \end{bmatrix}, \quad \begin{bmatrix} \alpha \\ \beta \end{bmatrix} > 0. \tag{11}$$

Note 3.2. For more abbreviation, denote $f = \left(\alpha^T, \beta^T\right)^T$, $b^s = \left(b^{lT}, b^{lT}\right)^T$, $\overline{A} = \begin{bmatrix} B^+ & -B^- \\ -B^- & B^+ \end{bmatrix}$, and for the vector x and positive vectors α, β define the corresponding fuzzy vector $\tilde{x} \in \mathbb{F}(\mathbb{R}^n)_{LR}$ by $\tilde{x} = (\tilde{x}_1, \ldots, \tilde{x}_n)^T$ with $\tilde{x}_i = (x_i, \alpha_i, \beta_i), 1 \leq i \leq n$.

Now, we give the fundamental theorem for the fuzzy LR linear system.

Theorem 3.3. *(Fundamental Theorem of FLRLS) Let $A \in \mathbb{R}^{m\times n}$ and $\tilde{b} \in \mathbb{F}(\mathbb{R}^m)$. Then, $x = (x_1, \ldots, x_n)^T \in \mathbb{R}^n$ and $f = \left(\alpha^T, \beta^T\right)^T$ are solutions of (10) and (11) respectively if and only if the corresponding $\tilde{x}$ is a solution of (4).*

Corollary 3.4. Let $\mathbb{S}_1$ be the set of solutions of (10) and $\mathbb{S}_2$ be the set of solutions of (11). Then, $\forall x \in \mathbb{S}_1, \forall f = \left(\alpha^T, \beta^T\right)^T \in \mathbb{S}_2$, the corresponding $\tilde{x}$ is a solution of (4).

3.1. **Solution Strategy.** According to the Fundamental Theorem of FLRLS (Theorem 3.3), for solving (4), we must solve systems (10) and (11). The system (10) is a crisp linear system and there are various methods for solving it. If the system (10) lacks solution, then the corresponding fuzzy system does not have any solution. To find the general solution of (10), we make use of the class of ABS algorithms [1]. Our reasons for using the ABS approach are:

(1) Not only ABS algorithms find all solutions of a crisp linear system but also Spedicato et al. in [21] reported that on some classes of significant problems, ABSPAK (a mathematical package using ABS algorithm) codes have better performances in both accuracy and speed than the codes in LAPACK. Therefore, the implementation of the proposed algorithm using the ABS approach is expected to be more effective than the extended approach introduced by Friedman et al. [10].

(2) We can obtain specific solutions by a proper choice of parameters of an ABS algorithm. For example, the unique solution with minimal Euclidean norm in the solution space (see Section 2.2) would be an interesting one, whenever more than one solution exist.

Proceeding with the solution of the system (4), after solving (10), we need to solve the system (11). In (11), we have $\overline{A} \geq 0$ and $b > 0$. So we must find positive solutions of a nonnegative crisp system. If $m = n$ and $\overline{A} > 0$ then $\overline{A}^{-1} > 0$ if only if $\overline{A}$ is a general permutation matrix [17]. Friedman et al. [10] gave an example with coefficient matrix $\overline{A} > 0$, $\overline{A}^{-1} \ngtr 0$ and $\overline{A}^{-1} b \ngtr 0$, for which a positive solution does not exist. Thus, they introduced the concept of weak solution of the fuzzy linear system.
Here, we introduce a new notion of fuzzy LR solution for (4). Suppose that $\tilde{x}$ is so that its mean value x is a solution of (10). Then, we define,

$$r_{\tilde{x}} = \left\| B^+\alpha - B^-\beta - b^l \right\|_2^2 + \left\| -B^-\alpha + B^+\beta - b^r \right\|_2^2 . \tag{12}$$

If $r_{\tilde{x}} = 0$, then $(\alpha^T, \beta^T)^T$ is a solution of (11), and hence $\tilde{x} = (x, \alpha, \beta)$ is a solution of (4). We refer to such a solution as an exact solution.

Definition 3.5. If x is so that $Ax = b$, α and β are so that (12) evaluates to be zero then the corresponding $\tilde{x} = (x, \alpha, \beta)_{LR}$ is said to be an exact solution of (4).

Note 3.6. Definition 3.5 is equivalent to the definition given by Friedman et al. [10], when A is restricted to a nonsingular matrix. In this case, one can compute an exact solution using the extension principle (see [10]).

But, if $r_{\tilde{x}} > 0$ for all $\tilde{x} \in \mathbb{F}(\mathbb{R}^n)_{LR}$, then the system (11) lacks a solution, and hence FLRLS lacks an exact solution. In this case, in analogy with the case of crisp least squares problem, it would be appropriate to specify a fuzzy LR vector as an approximate solution, using the minimizer of (12) as the spreads and the mean value of its components as a solution of the crisp system (10). We note that this solution would be, in general, different from the weak solution considered by

Friedman et al. [10] in the special nonsingular case. Thus, we have the following definition of an approximate solution.

Definition 3.7. If x is so that $Ax = b$, and $f = (\alpha^T, \beta^T)^T = \texttt{argmin}\{r_{\tilde{y}} | \tilde{y} \in \mathbb{F}(\mathbb{R}^n)_{LR}\}$ so that (12) yields to be nonzero, then the corresponding $\tilde{x}(x, \alpha, \beta)_{LR}$ is said to be an approximate solution of (4).

Note that in [10], $\tilde{x} = \overline{A}^{-1}\tilde{b}$ is computed first, and then it is checked whether $\tilde{x}$ is a fuzzy vector. If $\tilde{x}$ is not a fuzzy vector then a corresponding weak solution is constructed from $\tilde{x}$. Our approach here is different from [10]. We first solve (10). If (10) lacks a solution, then it is apparent that the original fuzzy system (4) does not have any solution. On the other hand, when (10) has a solution, then we find a fuzzy LR vector with a minimal value in (12). If the minimal value of (12) is zero then we have an exact solution; otherwise, we have an approximate solution. We realize that in our approach there is no need to check whether a solution is fuzzy, as required by the approach in [10].
We next introduce an algorithm for solving FLRLS, to compute exact or approximate solutions. We must find a fuzzy LR vector in $\mathbb{F}(\mathbb{R}^n)_{LR}$, say $\tilde{y}$, such that $r_{\tilde{y}}$ (as defined in (12)) is minimal and the mean value of components of $\tilde{y}$ is a solution of the crisp system (10). Thus, making use of the Fundamental Theorem of FLRLS (Theorem 3.3), after solving (10), we would need to solve the following constrained least squares problem,

$$\begin{cases} z = \min\left\{ \left\| B^+\alpha - B^-\beta - b^l \right\|_2^2 + \left\| -B^-\alpha + B^+\beta - b^r \right\|_2^2 \right\} \\ s.t. \\ \alpha = (\alpha_1, \ldots, \alpha_n)^T > 0, \beta = (\beta_1, \ldots, \beta_n)^T > 0. \end{cases}$$

Remark 3.8. For the above optimization problem, we may consider the numbers α_j, β_j for $j = 1, \ldots, n$ to have small positive values (e.g., 10^{-20}). In this case, the fuzzy numbers $(x_j, \alpha_j, \beta_j)_{LR}^T$ are very close to the crisp numbers x_j. Thus, the user can define a positive parameter, ε, such that $\alpha_j, \beta_j \geq \varepsilon$ for all j. By this setting, it is guaranteed that each fuzzy number has left and right spreads as much as ε (see Definition 2.6). Therefore, instead of the above optimization problem, we consider the following optimization problem,

$$\begin{cases} z = \min\left\{ \left\| B^+\alpha - B^-\beta - b^l \right\|_2^2 + \left\| -B^-\alpha + B^+\beta - b^r \right\|_2^2 \right\} \\ s.t. \\ \alpha = (\alpha_1, \ldots, \alpha_n)^T \geq \varepsilon e, \beta = (\beta_1, \ldots, \beta_n)^T \geq \varepsilon e, \end{cases} \tag{13}$$

where $e = (1, \ldots, 1)^T \in \mathbb{R}^n$ and $\varepsilon > 0$ is a user-defined parameter.

We observe that an exact solution exists if and only if (13) has a solution with $z^* = 0$ (in this case, a solution of (13) is a solution of the system (11)); otherwise, system (11) lacks an exact solution and thus approximate solutions can be computed for the FLRLS. In any case, a solution (approximate or exact) is determined by a corresponding $\tilde{x}$ composed of any solution of the crisp system (10) and the linear least squares problem (13).

3.1.1. **A Fuzzy Concept for LR Solutions.** It is possible to characterize the set of solutions of FLRLS as a fuzzy set. For this, we propose defining exact and approximate solutions based on a proposed membership function. We introduce a new notion of fuzzy LR solution for (4). Let $\tilde{y} = (y_i, \alpha_i, \beta_i)^T_{LR}, 1 \leq i \leq n$, where y is a solution of (10). Define the fuzzy set of LR solutions of (4) by the following membership function,

$$\mu_S(\tilde{y}) = \begin{cases} g(r_{\tilde{y}}), & 0 \leq r_{\tilde{y}} \leq u, Ay = b, \\ 0, & O.W., \end{cases} \tag{14}$$

where $\tilde{y} \in \mathbb{F}(\mathbb{R}^n)_{LR}$ and $r_{\tilde{y}}$ is as defined in (12), $u > 0$ is a user-defined parameter and $g : [0, u] \to [0, 1]$ is a strictly decreasing linear function with $g(0) = 1$ and $g(u) = 0$. An example function g may be:

$$g(r) = \begin{cases} 1 - \frac{r}{u}, & 0 \leq r \leq u, \\ 0, & O.W. \end{cases} \tag{15}$$

(We note that other appropriate functions g with the specified conditions (14) may also serve as membership functions, but we would make use of (15) in our experiments at the end of this section.)
In this definition, S, the set of LR solutions of (4), is a fuzzy set and each $\tilde{y} \in \mathbb{F}(\mathbb{R}^n)_{LR}$ is seen as a solution of FLRLS. Also, the solution set of FLRLS at the α-level is defined to be $S_\alpha = \{\tilde{x} | \mu_S(\tilde{x}) = \alpha\}$.

Note 3.9. $\tilde{x}$ is an exact solution of FLRLS if and only if $r_{\tilde{x}} = 0$ (see Definition 3.5), or equivalently $g(r_{\tilde{x}}) = 1$ or $\mu_S(\tilde{x}) = 1$.

Since $\mu_S(\tilde{y}) = 1$ implies that $\tilde{y}$ is an exact solution of FLRLS, then it would be appropriate to find solutions of FLRLS in $\mathbb{F}(\mathbb{R}^n)_{LR}$ with maximum membership function value. In other words, to solve FLRLS, we need to solve the following problem,

$$\max_{\tilde{y} \in \mathbb{F}(\mathbb{R}^n)_{LR}} \mu_S(\tilde{y}),$$

or equivalently,

$$\max_{\tilde{y} \in \mathbb{F}(\mathbb{R}^n)_{LR}} g(r_{\tilde{y}}),$$

where $r_{\tilde{y}}$ is defined as in (12). Next, we can redefine an exact solution of FLRLS in the following context.

Definition 3.10. $\tilde{x} \in \mathbb{F}(\mathbb{R}^n)_{LR}$ is an exact solution of FLRLS if $\mu_S(\tilde{x}) = 1$ (or equivalently $\tilde{x} \in S_1$).

It is now apparent that if $S_1 = \emptyset$, then FLRLS lacks an exact solution. In other words, for all $\tilde{y} \in \mathbb{F}(\mathbb{R}^n)_{LR}$ we have $r_{\tilde{y}} > 0$ (as defined in (12)), and so, the system (11) lacks an exact solution. In this case, it would be useful to find a fuzzy LR vector with the maximum μ_S value. We can thus characterize an approximate solution as being a fuzzy LR vector with the maximum μ_S value as follows.

Definition 3.11. We say that $\tilde{x}$ is an approximate solution of FLRLS if $\tilde{x} \in S_{\alpha_{\max}}$, where $\alpha_{\max} = \max\{\alpha | S_\alpha \neq \emptyset\}$. We call such an $\tilde{x}$ an approximate solution at the $\alpha_{\max}$-level.

3.1.2. **Finding General Solution of FLRLS.** We have seen that, for solving FLRLS, we need to solve a crisp linear system and a constrained least squares problem. Here, we propose the following algorithm to solve the FLRLS or equivalently find the general solution (either exact or approximate).

Algorithm 2 Find general solution of FLRLS.

Step 0: (Initialization) Give $u > 0$ for (15) and $\varepsilon > 0$ for (13).

Step 1: Find general solution of (10) by an ABS algorithm and let S_{ABS} be a set of all solutions of (10). **If** $S_{ABS} = \emptyset$ **then stop** (the fuzzy linear system (4) does not have any solution; see Definitions 3.10 and 3.11).

Step 2: Solve the optimization problem (13) and let (α, β) be an optimal solution of (13) with optimal value z^*.

Step 3:

If $z^* = 0$ **then**

FLRLS has exact solutions and

$$S = \left\{\tilde{x} \quad | \quad \tilde{x} = (\tilde{x}_i)_{LR}^T, \tilde{x}_i = (x_i, \alpha_i, \beta_i), 1 \leq i \leq n, x \in S_{ABS}\right\},$$

is the set of all exact solutions

else

FLRLS has approximate solutions and

$$S_{g(z^*)} = \left\{\tilde{x} \quad | \quad \tilde{x} = (\tilde{x}_i)_{LR}^T, \tilde{x}_i = (x_i, \alpha_i, \beta_i), 1 \leq i \leq n, x \in S_{ABS}\right\},$$

is the set of all $g(z^*)$-level solutions of FLRLS, where the function $g(x)$ is as defined in (15).

Step 4: Stop.

Remark 3.12. In system (10), $A_{m\times n}$, $m \leq n$, is a crisp matrix. If $rank(A) = m = n$, then system (10) has a unique solution. If, however, $rank(A) \leq m < n$ then system (10) may either lack a solution or have infinitely many solutions (whenever there is one solution). Using an ABS Algorithm, we can recognize that system (10) is compatible or not (see **Step** 1 in Algorithm 2 and **Step** 3 in Algorithm 1). If system (10) is incompatible, then based on Definitions 3.10 and 3.11, the fuzzy linear system (4) lacks any solution (either exact or approximate). But, if system (10) is compatible then, using Algorithm 1, we can obtain the general solution of system (10).

3.2. **Further Constraint Specifications.** Algorithm 2 is flexible enough to be modified for special applications easily. For example, suppose the user wants to impose α_j , $\beta_j \leq M_j$ or decides to have some LR fuzzy numbers to be approximately symmetric, then the user adjoins appropriate constraints to the constrained least squares problem (13); e.g., $|\alpha_j - \beta_j| \leq 10^{-k}$ for some j, $j \in \{1, \ldots, n\}$. Also, as mentioned in Remark 3.8, the user can control the right and left spreads of each

fuzzy number delicately by adding $\alpha_j, \beta_j \geq \varepsilon_j$ for some j, $j = 1, \ldots, n$, to (13). Other constraints can also be interesting. Assume the user wants the measure of fuzziness of some fuzzy numbers to be smaller than certain predefined positive numbers N_j. In this case, the user adds $\alpha_j + \beta_j \leq N_j$ to the constrained least squares problem (13).
So, one can change the constrained least squares problem in **Step** 2 of Algorithm 2 to:

$$\begin{cases} z = \min \left\{ \left\| B^+\alpha - B^-\beta - b^l \right\|_2^2 + \left\| -B^-\alpha + B^+\beta - b^r \right\|_2^2 \right\} \\ \quad s.t. \\ \qquad \alpha = (\alpha_1, \ldots, \alpha_n)^T \geq \varepsilon e, \beta = (\beta_1, \ldots, \beta_n)^T \geq \varepsilon e, \\ \qquad \alpha, \beta \in C, \end{cases} \tag{16}$$

where C is the feasible space defined by user's specified conditions. So, we can change **Step** 2 in Algorithm 2 to "Solve the optimization problem (16) and let (α, β) be an optimal solution of (16)".
To illustrate the effectiveness of our approach, we discuss the following two examples.

Example 3.13. [10] Consider,

$$\begin{cases} \tilde{x}_1 - \tilde{x}_2 = (1, 1, 1)_{LR} \\ \tilde{x}_1 + 3\tilde{x}_2 = (5, 1, 2)_{LR}. \end{cases}$$

Case 1: We solve the following two problems using $u = 4$ in (15) and $\varepsilon = 10^{-2}$ in (13) or (16):

$$\begin{cases} x_1 - x_2 = 1, \\ x_1 + 3x_2 = 5, \end{cases} \quad and \quad \begin{cases} z = \min \Big[(\alpha_1 + \beta_2 - 1)^2 + (\beta_1 + \alpha_2 - 1)^2 \\ \qquad + (\alpha_1 + 3\alpha_2 - 1)^2 + (\beta_1 + 3\beta_2 - 2)^2 \Big] \\ s.t. \\ \alpha_j, \beta_j \geq \varepsilon, \quad j = 1, 2. \end{cases}$$

Then, we obtain $x_1 = 2, x_2 = 1$ and $z^* = 0$ with $\alpha_1 = 0.625, \beta_1 = 0.875, \alpha_2 = 0.125, \beta_2 = 0.375$. Thus,$\tilde{x} = (\tilde{x}_1, \tilde{x}_2)^T_{LR}$ with $\tilde{x}_1 = (2, 0.625, 0.875)_{LR}$, $\tilde{x}_2 = (1, 0.125, 0.375)_{LR}$ is an exact solution.

Case 2: Consider desiring $\alpha_1 \simeq \beta_1$ and $\alpha_2 \simeq \beta_2$ in the sense of $|\alpha_1 - \beta_1| \leq 10^{-2}$ and $|\alpha_2 - \beta_2| \leq 10^{-2}$. Thus, we add these inequalities to the constrained least squares problem. Then, the global optimum is computed to be: $x_1 = 2, x_2 = 1, \alpha_1 = 0.745, \beta_1 = 0.755, \alpha_2 = 0.245, \beta_2 = 0.255$ and $z^* = 0.4608$. Therefore, the approximate solution is a (0.8848)-level solution $\tilde{x} = (\tilde{x}_1, \tilde{x}_2)^T_{LR}$ with $\tilde{x}_1 = (2, 0.745, 0.755)_{LR}$ and $\tilde{x}_2 = (1, 0.245, 0.255)_{LR}$, (with $u = 4$ in (15)).

Case 3: Consider desiring $\alpha_1 = \beta_1$ and $\alpha_2 = \beta_2$, by adding them as constraints. Then, the global optimum is computed as: $x_1 = 2, x_2 = 1, \alpha_1 =$

$\beta_1 = 0.75, \alpha_2 = \beta_2 = 0.25$ and $z^* = 0.5$. Therefore, the approximate solution is a (0.875)-level solution with $\tilde{x} = (\tilde{x}_1, \tilde{x}_2)^T_{LR}$, $\tilde{x}_1 = (2, 0.75, 0.75)_{LR}$ and $\tilde{x}_2 = (1, 0.25, 0.25)_{LR}$, (with $u = 4$ in (15)).

Remark 3.14. Cases 2 and 3 in Example 3.13 show that if we desire $\alpha_1 \simeq \beta_1$ and $\alpha_2 \simeq \beta_2$ then we can achieve an approximate solution with a higher α-level of solution as compared with the case $\alpha_1 = \beta_1$ and $\alpha_2 = \beta_2$.

Example 3.15. Consider,

$$\begin{cases} \tilde{x}_1 + \tilde{x}_2 + \tilde{x}_3 = (1, 1, 1)_{LR} \\ \tilde{x}_1 + \tilde{x}_2 - \tilde{x}_3 = (2, 1, 1)_{LR}. \end{cases}$$

Using Algorithm 2 (with $u = 4$ in (15) and $\varepsilon = 10^{-2}$ in (13) or (16)), we obtain $x = \begin{bmatrix} 0.75 \\ 0.75 \\ -0.5 \end{bmatrix} + \begin{bmatrix} 0.5 \\ -0.5 \\ 0 \end{bmatrix} * q$,where $q \in \mathbb{R}$, is general solution of (10). One optimal solution of (13) is $\alpha_j = \beta_j = 0.3333, j = 1, 2, 3$. Then, $\tilde{x} = \begin{bmatrix} (0.75 + 0.5q, 0.3333, 0.3333)_{LR} \\ (0.75 - 0.5q, 0.3333, 0.3333)_{LR} \\ (-0.5, 0.3333, 0.3333)_{LR} \end{bmatrix}$, for all $q \in \mathbb{R}$, is a symmetric exact solution of FLRLS. In a similar way,$\tilde{x} = \begin{bmatrix} (0.75 + 0.5q, 0.4698, 0.3556)_{LR} \\ (0.75 - 0.5q, 0.2415, 0.3556)_{LR} \\ (-0.5, 0.2887, 0.2887)_{LR} \end{bmatrix}$, for all $q \in \mathbb{R}$, is another exact solution.

Remark 3.16. If in (16), we select $\varepsilon = 0$, then it is possible that the fuzzy system (4) has solution with a higher α-level of solution as compared with the case, $\varepsilon > 0$. Of course, in this case ($\varepsilon = 0$ in (16)), solution of (4) may not be a fuzzy LR vector as shown by the following example.

Example 3.17. Consider,

$$\begin{cases} \tilde{x}_1 + 3\tilde{x}_2 = (1, 1, 1)_{LR} \\ 7\tilde{x}_1 - 10\tilde{x}_2 = (7, 7, 7)_{LR}. \end{cases}$$

Case 1: We solved the above fuzzy LR system using $u = 4$ in (15) and $\varepsilon = 0$ in (13) or (16). Then, we obtained $x_1 = 1, x_2 = 0$ and $z^* = 0$ with $\alpha_1 = 1, \beta_1 = 1, \alpha_2 = 0, \beta_2 = 0$. Thus, $\tilde{x} = (\tilde{x}_1, \tilde{x}_2)^T_{LR}$ with $\tilde{x}_1 = (1, 1, 1)_{LR}$, $\tilde{x}_2 = 0 = (0, 0, 0)$ is an exact solution.

Case 2: We solved the above fuzzy LR system using $u = 4$ in (15) and $\varepsilon = 10^{-1}$ in (13) or (16). Then, we obtained $x_1 = 1, x_2 = 0$ and $z^* = 0.0484$ with $\alpha_1 = 0.854, \beta_1 = 0.854, \alpha_2 = 0.1, \beta_2 = 0.1$. Thus, the fuzzy LR vector $\tilde{x} = (\tilde{x}_1, \tilde{x}_2)^T_{LR}$ with $\tilde{x}_1 = (0, 0.854, 0.854)_{LR}$, $\tilde{x}_2 = (0, 0.1, 0.1)_{LR}$ is a (0.9879)-level solution (in this case, fuzzy system lacks an exact solution).

Remark 3.18. System (9) or (11), or correspondingly the constrained least squares problem (13), may have more than one solution (e.g., when m is smaller than n).

In such a case, the user may be interested in a particular solution by defining some constrains in (13). An interesting solution is one with the least Euclidean norm in the solution space (this solution is unique [14]). By this selection, we choose a solution of system (11) or equivalently the optimization problem (13) that has the least measure of fuzziness in the fuzzy solution space. To illustrate this, see the following example.

Example 3.19. Consider,

$$\begin{cases} \tilde{x}_1 - 2\tilde{x}_2 + 3\tilde{x}_3 - \tilde{x}_4 = (5, 5, 4.5)_{LR} \\ \tilde{x}_1 + \tilde{x}_2 - \tilde{x}_3 + \tilde{x}_4 = (-1, 3, 3.25)_{LR}. \end{cases}$$

Using Algorithm 2 (with $u = 4$ in (15) and $\varepsilon = 10^{-2}$ in (13), or (16)), we obtain:

$$x = \begin{bmatrix} 0.7143 \\ -0.5714 \\ 1.0000 \\ -0.1429 \end{bmatrix} + \begin{bmatrix} 0.1714 & -0.0571 & -0.2000 & -0.3143 \\ -0.0571 & 0.6857 & 0.4000 & -0.2286 \end{bmatrix}^T * q,$$

where $q \in \mathbb{R}^2$ is the general solution of (10); this solution is obtained from the modified Huang's algorithm, as pointed out in Section 2.2, and the solution of constrained least squares problem[1] with 'lsqlin' function of MATLAB software. The optimal solution of (13) with the least Euclidean norm in the solution space is: $(\alpha^T, \beta^T)^T$,where $\alpha = [\ 0.9701,\ 0.6766,\ 0.5045,\ 0.9487\ ]^T$ and $\beta = [\ 0.9844,\ 0.7552,\ 0.4045,\ 1.0058\ ]^T$. Thus,

$$\tilde{x} = \begin{bmatrix} (+0.7143, 0.9701, 0.9844)_{LR} \\ (-0.5714, 0.6766, 0.7552)_{LR} \\ (+1.0000, 0.5045, 0.4045)_{LR} \\ (-0.1429, 0.9487, 1.0058)_{LR} \end{bmatrix}$$

is the unique exact solution of FLRLS with the least Euclidean norm in the solution space (with the Euclidean norm of mean values equal to 1.3628 and the Euclidean norm of spread values equal to 2.2953). This problem has more than one exact solution. The other solutions have higher Euclidean norms. It can be seen that

$$\tilde{x} = \begin{bmatrix} (0.5, 0.1550, 1.6250)_{LR} \\ (0.0, 0.0100, 0.0100)_{LR} \\ (1.5, 1.6050, 0.0100)_{LR} \\ (0.0, 2.8250, 0.0100)_{LR} \end{bmatrix}$$

is another exact solution with the Euclidean norm of mean values equal to 1.5811 and the Euclidean norm of spread values equal to 3.6362 (this solution is obtained using '\' command and 'lsqlin' function of MATLAB software).

Remark 3.20. (Duality in fuzzy linear systems) Usually, there is no inverse element for an arbitrary fuzzy LR number. Also, if $\tilde{x} \in \mathbb{F}(\mathbb{R}^1)_{LR}$ is an arbitrary LR fuzzy number, then we have,

$$\tilde{x} \bigoplus (-\tilde{x}) \neq 0.$$

[1] $\min \{f^T f | \bar{A} f = b^s, f \geq \varepsilon e\}$; see Note 3.2.

Therefore, the following two fuzzy linear systems are not equivalent:

$$A_1\tilde{x}+\tilde{b}_1=A_2\tilde{x}+\tilde{b}_2, \qquad (A_1-A_2)\tilde{x}=(\tilde{b}_2-\tilde{b}_1).$$

In [13], the necessary and sufficient conditions for $A_1\tilde{x}=A_2\tilde{x}+\tilde{b}_2$ are given, where A_1 and A_2 are square matrices. With a similar argument as in the proof of Theorem 3.3, using the same notation, we can prove the following theorem.

Theorem 3.21. *Suppose* $A_1=\left[a_{ij}^1\right], A_2=\left[a_{ij}^2\right]\in\mathbb{R}^{m\times n}$ *and* $\tilde{b}_1,\tilde{b}_2\in\mathbb{F}(\mathbb{R}^m)_{LR}$ *are given. Then,* $\tilde{x}=(\tilde{x}_i),\tilde{x}_i=(x_i,\alpha_i,\beta_i),1\leq i\leq n$, *is a solution of* $A_1\tilde{x}+\tilde{b}_1=A_2\tilde{x}+\tilde{b}_2$ *if and only if* $x=(x_1,\ldots,x_n)^T\in\mathbb{R}^n$ *is a solution of* $(A_1-A_2)x=(b_2-b_1)$ *and* $\left(\alpha^T,\beta^T\right)^T$ *is a positive solution of the following system,*

$$\begin{cases}\begin{bmatrix}B_1^+-B_2^+ & -B_1^-+B_2^-\\ -B_1^-+B_2^- & B_1^+-B_2^+\end{bmatrix}\begin{bmatrix}\alpha\\ \beta\end{bmatrix}=\begin{bmatrix}b_2^l-b_1^l\\ b_2^r-b_1^r\end{bmatrix},\\ \alpha,\beta>0,\end{cases}$$

where,

$$[B_k^+]_{ij}=\begin{cases}a_{ij}^k, & a_{ij}^k\geq 0,\\ 0, & a_{ij}^k<0,\end{cases}\qquad [B_k^-]_{ij}=\begin{cases}a_{ij}^k, & a_{ij}^k<0,\\ 0, & a_{ij}^k\geq 0,\end{cases}$$

for all $k=1,2$, $i=1,\ldots,m$, $j=1,\ldots,n$, $b_k^l=(b_1^{l,k},\ldots,b_m^{l,k})^T$ and $b_k^r=(b_1^{r,k},\ldots,b_m^{r,k})^T$, for $k=1,2$.

Analogous to our previous arguments, suppose $\tilde{y}=(\tilde{y}_i),\tilde{y}_i=(y_i,\alpha_i,\beta_i),1\leq i\leq n$, let $b^l=b_2^l-b_1^l$, $b^r=b_2^r-b_1^r$ and

$$r_{\bar{y}}=\left\|(B_1^+-B_2^+)\alpha+(-B_1^-+B_2^-)\beta-b^l\right\|_2^2+\left\|(-B_1^-+B_2^-)\alpha+(B_1^+-B_2^+)\beta-b^r\right\|_2^2.$$

Now, define the fuzzy set of solutions of $A_1\tilde{x}+\tilde{b_1}=A_2\tilde{x}+\tilde{b_2}$ by the following membership function,

$$\mu_S(\tilde{y})=\begin{cases}g(r_{\bar{y}}), & 0\leq r_{\bar{y}}\leq u,(A_1-A_2)y=b_1-b_2,\\ 0, & O.W.,\end{cases}$$

where $u>0$ is a user-defined parameter and g is defined as in (15).
Let $S_\alpha=\{\tilde{x}|\mu_S(\tilde{x})=\alpha\}$. Then, $\tilde{x}$ is an exact solution of $A_1\tilde{x}+\tilde{b_1}=A_2\tilde{x}+\tilde{b_2}$ if and only if $\tilde{x}\in S_1$. If $S_1=\emptyset$, then no solution exists and $S_{\alpha_{max}}$ is a set of approximate solutions, where $\alpha_{max}=\max\{\alpha|S_\alpha\neq\emptyset\}$. The solution strategy for solving the system $A_1\tilde{x}+\tilde{b_1}=A_2\tilde{x}+\tilde{b_2}$ is similar to the one given in Section 3.1.

4. Conclusions

We presented a methodology for characterization and an approach for computing the solutions of fuzzy linear systems with LR fuzzy variables. Notions of exact and approximate solutions were characterized. We transformed the fuzzy linear system to a corresponding linear crisp system and a constrained least squares problem. If the corresponding crisp system is incompatible, then the fuzzy LR system lacks an exact solution. We showed that the fuzzy LR system had an exact solution if and

only if the corresponding crisp system would be compatible (had a solution) and the solution of the corresponding least squares problem would equal zero. In this case, the exact solution was determined by the solutions of the two corresponding problems. On the other hand, if the corresponding crisp system was compatible and the optimal value of the corresponding constrained least squares problem was nonzero, then we identified approximate solutions by solving the least squares problem. Also, we characterized solutions by defining a membership function so that an exact solution was a fuzzy LR vector with the value of membership function equal to one and, when an exact solution did not exist, an approximate solution was a fuzzy LR vector with a maximal membership function value. We proposed a class of algorithms based on ABS algorithm Δ for solving these systems. The proposed algorithms can also be used to solve the extended dual fuzzy linear systems. Finally, we showed that, when the system had more than one solution, the proposed algorithms were flexible enough to compute special solutions of interest. Various scenarios for solutions were demonstrated by working out several examples.

Acknowledgements. The authors thank the Research Council of Sharif University of Technology for its support. Also, the authors are grateful to the two anonymous referees and the editor for their constructive comments and questions. Consideration of all improved the presentation.

References

[1] J. Abaffy, C. Broyden and E. Spedicato, *A class of direct methods for linear systems*, Numerische Mathematik, **45** (1984), 361-376.

[2] J. Abaffy and E. Spedicato, *ABS projection algorithms: mathematical techniques for linear and nonlinear equations*, Ellis Horwood, Chichester, 1989.

[3] S. Abbasbandy and M. Alavi, *A method for solving fuzzy linear system*, Iranian Journal of Fuzzy Systems, **2** (2005), 37-43.

[4] F. Abramovich, M. Wagenknecht and Y. I. Khurgin, *Solution of LR-type fuzzy systems of linear algebraic equations*, Busefal, **35** (1988), 86-99.

[5] R. Bulrisch and J. Stoer, *Introduction to numerical analysis (texts in applied mathematics)*, 2nd edition, Springer, 1992.

[6] J. J. Buckley and Y. Qu, *Solving systems of linear fuzzy equations*, Fuzzy Sets and Systems, **43** (1991), 33-43.

[7] D. Dubois and H. Prade, *Fuzzy sets and systems theory and applications*, Academic Press, New York, 1980.

[8] E. Egervary, *On rank-diminishing operations and their applications to the solution of linear equations*, ZAMP, **9** (1960), 376-386.

[9] H. Esmaeili, N. Mahdavi-Amiri and E. Spedicato, *A class of ABS algorithms for Diophantine linear systems*, Numerische Mathematik, **91** (2001), 101-115.

[10] M. Friedman, M. Ma and A. Kandel, *Fuzzy linear systems*, Fuzzy Sets and Systems, **96** (1998), 201-209.

[11] R. Fuller, *On stability in possibilistic linear equality systems with Lipschitzian fuzzy numbers*, Fuzzy Sets and Systems, **34** (1990), 347-353.

[12] M. Khorramizadeh and N. Mahdavi-Amiri, *Integer extended ABS algorithms and possible control of intermediate results for linear Diophantine systems*, 4OR, **7(2)** (2009), 145- 167.

[13] M. Ma, M. Friedman and A. Kandel, *Duality in fuzzy linear systems*, Fuzzy Sets and Systems, **109** (2000), 55-58.

[14] G. H. Golub and C. F. Van Loan, *Matrix computations*, Baltimore, MD, 3rd edition, Johns Hopkins University Press, 1996.

[15] N. Mahdavi-Amiri and S. H. Nasseri, *Duality results and a dual simplex method for linear programming problems with trapezoidal fuzzy variables*, Fuzzy Sets and System, **158** (2007), 1961-1978.
[16] H. R. Maleki, M. Tata and M. Mashinchi, *Linear programming with fuzzy variables*, Fuzzy Sets and Systems, **109** (2000), 21-33.
[17] H. Minc, *Nonnegative matrices*, Wiley, New York, 1988.
[18] S. Muzzioli and H. Reynaerts, *Fuzzy linear systems of form* $A_1x + b_1 = A_2x + b_2$, Fuzzy Sets and Systems, **157** (2006), 939-951.
[19] H. T. Nguyen and E. A. Walker, *A first course in fuzzy logic*, Chapman & Hall, 2000.
[20] S. Nicolai and E. Spedicato, *A bibliography of the ABS methods*, Optimization Methods and Software, **8** (1997), 171-183.
[21] E. Spedicato, E. Bodon, A. D. Popolo and N. Mahdavi-Amiri, *ABS methods and ABSPACK for linear systems and optimization: a review*, 4OR, **1** (2003), 51-66.
[22] E. Spedicato, Z. Xia and L. Zhang, *ABS algorithms for linear equations and optimization*, Journal of Computational and Applied Mathematics, **124** (2000), 155-170.
[23] A. Vroman, G. Deschrijver and E. Kerre, *Solving systems of linear fuzzy equations by parametric functions-an improved algorithm*, Fuzzy Sets and Systems, **158** (2007), 1515 -1534.
[24] X. Wang, Z. Zhong and M. Ha, *Iteration algorithms for solving a system of fuzzy linear equations*, Fuzzy Sets and Systems, **119** (2001), 121-128.
[25] H. J. Zimmermann, *Fuzzy set theory and its applications*, Third ed., Kluwer Academic, Norwell, 1996.

Reza Ghanbari, Department of Mathematics, Ferdowsi University of Mashhad, Mashhad, Iran
E-mail address: rghanbari@matr.um.ac.ir

Nezam Mahdavi-Amiri*, Faculty of Mathematical Sciences, Sharif University of Technology, Tehran, Iran
E-mail address: nezamm@sharif.edu

Rohollah Yousefpour, Department of Mathematics, Mazandaran University, Babolsar, Iran
E-mail address: yosefpoor@mehr.sharif.edu

*Corresponding author